This research was supported by the Biological Sciences Division, AFOSR, SRLA under Contract/Grant AF 49(638)-917

BULLETIN OF
MATHEMATICAL BIOPHYSICS
VOLUME 24, 1962

CHURCH'S THESIS AND ITS RELATION TO THE CONCEPT OF REALIZABILITY IN BIOLOGY AND PHYSICS*

ROBERT ROSEN
COMMITTEE ON MATHEMATICAL BIOLOGY
THE UNIVERSITY OF CHICAGO

An attempt to characterize the physical realizability of an abstract mapping process in terms of the Turing computability of an associated numerical function is described. Such an approach rests heavily on the validity of Church's Thesis for physical systems capable of computing numerical functions. This means in effect that one must investigate in what manner Church's Thesis can be converted into an assertion concerning the nonexistence of a certain class of physical processes (namely, those processes which are capable of calculating the values of numerical functions which are not Turing-computable). A formulation which may be plausible is suggested, and it is then shown that the truth of Church's Thesis in this form is closely connected with the "effectiveness" of theoretical descriptions of physical systems. It is shown that the falsity of this form of Church's Thesis is related to a fundamental incompleteness in the possibility of describing physical systems, much like the incompleteness which Gödel showed to be inherent in axiomatizations of elementary arithmetic. Various implications of these matters are briefly discussed.

I. *General Introduction*. In previous work (Rosen, 1958a, 1958b, 1959) we have attempted to develop an abstract approach to biological systems, in which these systems are represented by families of mappings satisfying certain natural conditions. More recently, we have shown (Rosen, 1962) that the properties of the set of all such systems depend very strictly on the category from which we are allowed to choose the mappings which enter into our representation. Each category may, in fact, be said to give rise to its own "abstract biology."

*This research was supported by the United States Air Force through the Air Force Office of Scientific Research of the Air Research and Development Command, under Contract No. AF 49(638)-917.

The manner in which such "abstract biologies" depend on the structure of the underlying category is an interesting problem in itself. The question also arises, however, as to how to characterize the category giving rise to an abstract biology which most nearly resembles our own organic world in its behavior. One approach to this problem, which seems most natural, is to attempt to restrict ourselves only to those mappings which can in some sense be said to be "physically realizable." That is, we should try to find a means of characterizing those abstract mappings which can be mirrored by the action of a real physical system. It is clear that in an arbitrary category selected at random we will find a large number of mappings which cannot correspond to actual physical processes, and which will hence give rise to large numbers of systems which satisfy the axioms for $(\mathfrak{M},\mathfrak{R})$-systems, but which are not capable of actual existence. The properties of such systems are of no interest for purely biological studies, and in the absence of a means of distinguishing these systems from the "realizable" ones we may easily be misled by their behavior.

The present paper was developed out of an attempt to find a criterion which will allow us to characterize the "realizable" mappings in a category, and thus allow us to restrict ourselves at the outset to the subcategory of "realizable" mappings. We will then be sure that all conclusions derivable from the structure of the subcategory of realizable mappings will be of direct applicability to biological problems, and we may even hope to determine the relevant structural facts concerning such a category in a relatively direct manner.

The most natural approach to take seems to be the following: let $f: A \longrightarrow B$ be an arbitrary mapping. If f is to be physically realizable, it is no restriction to take A and B to be countable sets. Let Z denote the set of integers, and let definite bijective mappings $\varphi: Z \longrightarrow A$, $\psi: B \longrightarrow Z$ be chosen. Then we can write the following diagram:

$$
\begin{array}{ccc}
A & \xrightarrow{f} & B \\
\varphi \Big\uparrow & & \Big\downarrow \psi \\
Z & \xrightarrow[\bar{f}]{} & Z
\end{array}
\qquad (1)
$$

Here $\bar{f}: Z \longrightarrow Z$ is the composite mapping $\psi f \varphi$. If the mapping f is physically realizable, and if the mappings φ, ψ are in any sense "effective" (i.e., if we can "code" the integers into the objects A which serve as inputs to an actual physical realization of f, and "decode" them from B), then we can regard the physical process corresponding to the composite mapping $\psi f \varphi$ as a computing machine, and the numerical mapping $\bar{f}$ should therefore be a recursive or computable function. Conversely, it seems natural to try to turn this argument around, and assert that a mapping f should be physically realizable if and only if there exist "effective" mappings φ, ψ such that the induced mapping $\bar{f} = \psi f \varphi$ is a recursive function. We can then argue that if f were physically realizable and $\bar{f}$ were not recursive, we would be in the position that there would exist a physical process (namely the one corresponding to the composite map $\psi f \varphi$) which would effectively calculate the values of a nonrecursive function. This immediately seems to contradict Church's Thesis, which asserts precisely that the concepts of effective calculability and recursiveness are coextensive.

However, further reflection will reveal that the invoking of Church's Thesis to characterize a mapping f as realizable in terms of the recursiveness of the induced mapping $\bar{f}$ is not immediately justifiable. This is because Church's Thesis, as commonly enunciated, has no real physical content, although one is often read into it indirectly. In order to apply an argument of this type to problems of realizability, it is necessary to restate Church's Thesis as a physical proposition about the nonexistence of a certain class of physical processes. If we do this, we shall find that the physical form of Church's Thesis will be true only if the totality of physical laws governing the change of state of physical systems satisfies rather strong conditions. We shall be concerned in the present paper with describing the conditions which must hold in order for Church's Thesis to be a true statement.

The essence of our approach to this question is the following: given any physical process f which can act as a computing machine in the sense of the diagram (1), we shall show that we can in principle determine the algorithm (i.e., the "program" which the process is using to compute the values of $\bar{f}$) from the physical laws which govern the physical behavior of the process f, *if the description of the behavior of* f *in terms of the laws of physics is sufficiently sharp.* If there can exist systems for which the requi-

site sharpness cannot be attained, then Church's Thesis as a physical proposition need not be true, and hence a fortiori we cannot hope to characterize realizability in these terms.

It might be of some interest to point out that the essence of our approach to this problem, as described in the preceding paragraph, is the reverse of the type of argument which is employed in Relational Biology. We generally use relational arguments to infer a physical structure (or more accurately, a class of physical structures which are in some sense equivalent to each other) from a relational structure, i.e., from some kind of "program." Our investigation of the relation of Church's Thesis to questions of realizability involves the attempt to deduce the "program" employed in a physical process from the details of its physical structure, and is therefore primarily (cf. Rashevsky, 1954) of a Metric nature. Thus these remarks may be pertinent to the clarification of the Metric and Relational aspects of the structure of science (cf. Rashevsky, *loc. cit.*; Rosen, 1961).

II. *Preliminary Remarks.* The conjecture that every numerical function which is in any sense "effectively calculable" must also be recursive or computable (in the sense of Turing) is called Church's Thesis (Church, 1936; Kleene, 1952). As we have pointed out above, we shall find that apart from its purely logical connotations, this Thesis may be interpreted as asserting that a certain class of physical processes (namely, those capable of calculating the values of a function which is not Turing-computable) cannot exist. With this interpretation, Church's Thesis may legitimately be considered as a proposition in pure physics. The physical aspect of Church's Thesis is in fact implicit in many discussions of computability (cf., for example, the rhetorical question of Davis (1958): "...how can we ever exclude the possibility of our being presented someday (perhaps by some extra-terrestrial visitor) with a device... that "computes" a non-computable function?"). Despite this, no explicit discussion of the consequences which arise from a physical interpretation of Church's Thesis has been forthcoming, as far as the author is aware.

Church's Thesis remains a conjecture in Mathematical Logic because the notion of effective calculability is an informal and qualitative one, and it is not possible to prove formally that a quantitative concept, such as recursiveness, or Turing-computa-

bility, is broad enough to include all the aspects which could appropriately be included in the intuitive term "effectively calculable." If we attempt to interpret Church's Thesis as a physical proposition, however, it becomes necessary to give a physical characterization of what we will mean in an operational sense by the term "effectively calculable." When we do this, Church's Thesis will, as a physical proposition, take on a truth value, depending on two things: the operational interpretation we give to the term "effectively calculable," and the rules or laws we suppose to hold in the physical world. It may not be out of place to remark that this type of procedure (i.e., the interpretation of purely mathematical assertions in physical terms) is a well-established and often highly fruitful practice. It was precisely an investigation of this type, for example, which led from the dissertation of Riemann (1892), which dealt with purely geometric problems, to the development of General Relativity.

Therefore, we shall proceed in the next section to provide an operational interpretation of "effective calculability." In Section IV we review briefly some well-known aspects concerning the description of physical processes, and in Section V we shall show that, contingent upon the assumption of certain hypotheses concerning the "completeness" of physical descriptions, we can actually prove that Church's Thesis follows from the laws of physics. We shall conclude with a brief discussion of this result, and the mention of some of its implications for physical and biological systems.

We shall now introduce some terminology which will be used in the sequel. We shall call any physical entity or construct which is capable of calculating the values of a numerical function (in a sense to be made precise below) a *calculating agent* (or simply *agent*). The various inputs and outputs to such an agent will be called *objects*. The composite structure consisting of an agent and an appropriate input object will be called a *calculating system*.

III. *A Physical Characterization of Effective Calculability.* In this section, we undertake to specify the conditions which a physical entity f must satisfy in order to be said to operate as a calculating agent. We shall motivate our discussion by a preliminary examination of the special class of agents known as Turing ma-

chines, when these are considered as real physical mechanisms rather than as *gedankenmaschinen*. This can always be done; Turing himself (1937) pointed out that he considered only computing devices that could *in principle* be physically realized.

Parenthetically, this seems to be an appropriate place to remark that by the phrase "in principle" we do *not* mean "practicably." To dispel any possibility of misunderstanding, we may point out that there exist in the literature (cf. for example Bremermann, 1962) arguments which show that the laws of physics place certain upper limits on practicability. For example, it can be shown that certain simply statable combinatorial problems, which are finite and hence "trivially" solvable by search through all possibilities, could not in fact have been solved by a device as large as the earth operating at theoretically maximum rate for a time equal to the age of the earth. Arguments of this type have no bearing upon the scope and intent of our present discussion; this should be kept in mind as we proceed, especially in Section V.

Let us then suppose than an abstract Turing machine T is given, specified by the alphabet a_0 (the empty symbol), $a_1, \ldots, a_n$ and the set of internal configurations $q_0, q_1, \ldots, q_m$. The first step in the transition from the abstract machine T to a physical realization of T is to represent the symbols of the alphabet by a set of $n+1$ mutually distinguishable physical objects, which will be capable of serving as inputs to the machine. (In most contexts, it is customary to refer to an input to a Turing machine as a *sequence* of symbols of the alphabet. For our purposes it is convenient to regard the symbols themselves as inputs, which will henceforth be referred to as *elementary inputs*; sequences of such symbols will be called *compound inputs*).

In the realization envisioned by Turing, the physical objects chosen to represent the abstract elementary inputs are certain distinguished black marks on a tape. But it is clear that we are at liberty to use *any* family A of $n+1$ mutually distinguishable physical objects as elementary inputs to an appropriately constructed machine. We remark explicitly that we here use the term "physical object" in a wide sense; for example, we may consider different electromagnetic impulses as constituting legitimate inputs to a Turing machine, and hence as physical objects.

The abstract Turing machine T performs elementary acts by transforming the various elementary inputs a_i into new symbols,

depending on the initial internal configuration of the machine. In Turing's original formulation, each symbol of the alphabet is to be transformed into another symbol of the *same* alphabet. However, a moment's reflection will show that this restriction is not essential; that is, there is no necessity that what we may call the output alphabet of the machine T must consist of the same symbols as the original input alphabet. All that is necessary here is that the set of objects into which a Turing machine can transform the various elementary inputs shall consist of $n + 1$ mutually distinguishable objects, and in fact *any* set B of $n + 1$ such objects can serve as the output alphabet for an appropriately constructed Turing machine, defined on an appropriately selected set A of elementary inputs.

The interesting aspect of the Turing machine is not what it does to elementary inputs, but what it does to compound inputs. The well-known behavior of abstract Turing machines on such inputs may be stated formally in the following manner: for each internal state q_j of the machine, there is an associated mapping Φ_j, the domain of which is the set of sequences of symbols of the input alphabet A, and the range of which is the set of sequences of symbols of the output alphabet B. That is, Φ_j maps the free semigroup $G(A)$, generated by the symbols of A, into the free semigroup $G(B)$ generated by the symbols of B. *In this manner, we have made it possible for a given Turing machine* T *to be defined on an infinite set of inputs* (namely $G(A)$) instead of only on the finite set A.

We come now to the essential point in our discussion of the Turing machine. In order for such a machine to carry out *numerical* computations, it is obviously necessary that some effective means be at hand whereby the various compound inputs and outputs of the machine may be unambiguously associated with integers. In greater detail, it is necessary that, given any integer n, we shall be able to construct the word in $G(A)$ which is to correspond to n; moreover, to each word in $G(B)$ we must be able to associate the unique integer which is to correspond to this word. That is, we must be given *in advance* and in an effective manner the following pair of (1, 1) mappings:

$$\varphi : Z \longrightarrow G(A)$$

$$\psi : G(B) \longrightarrow Z$$

where Z denotes the set of integers. The operation of the Turing machine T as a computer of numerical functions may then be symbolized by the following diagram of mappings:

$$\begin{tikzcd} G(A) \arrow[r, "\Phi_j"] & G(B) \arrow[d, "\psi"] \\ Z \arrow[u, "\varphi"] \arrow[r, "\overline{\Phi}_j"'] & Z \end{tikzcd}$$

Here $\overline{\Phi}_j$ is the numerical function calculated by the machine T when the initial configuration of T is q_j; explicitly, we have $\overline{\Phi}_j(n) = \psi\, \Phi_j\, \varphi(n)$, whenever the right-hand side of this equation is defined.

Thus, we see that a Turing machine T, considered as a *physical* agent, is specified as a calculating machine by the following data:

1. A physical agent Φ_j (we shall denote the physical structure and the associated mapping by the same symbol) which transforms the objects of a certain infinite set $G(A)$ into the objects of another infinite set $G(B)$.
2. A prescribed effective mapping $\varphi: Z \longrightarrow G(A)$.
3. A similar mapping $\psi: G(B) \longrightarrow Z$.

These data are in fact sufficient to characterize the agent Φ_j as capable of evaluating a numerical function. It is immediate that every numerical function (such as $\overline{\Phi}_j$) which is induced in the above manner is Turing-computable. Moreover, it is readily verified that, given a Turing-computable numerical function, it is possible to find a physical agent and mappings φ, ψ, such that the given function is induced in the above manner by this agent.

We may now generalize the above consideration. Intuitively, we recognize that in order for any physical construct or entity to function as a calculating agent, it is necessary to be able to somehow "code" the set of integers into the inputs and outputs of this entity. That is, the data 2 and 3 characterizing the Turing machine cannot be essentially generalized. However, we may observe the following: in the case of the Turing machine, the effectiveness of the mappings φ, ψ is a consequence of the fact that the sets $G(A)$ and $G(B)$ of inputs and outputs, respectively, are finitely generated free semigroups. This consideration suggests that we may generalize the Turing machine by discarding the

assumption that the sets of input and output objects of a proposed calculating agent bear this type of algebraic structure, and retaining only the hypothesis that the maps φ, ψ are effective.

We are thus led to the following characterization of the most general calculating agent: A physical entity f will be called a calculating agent if and only if the following conditions hold:

4. There exist two (not necessarily distinct) countably infinite sets, denoted by $d(f)$ (the domain of f) and $r(f)$ (the range of f), such that the objects in $d(f)$ serve as inputs to f and the objects in $r(f)$ serve as outputs of f.
5. There exists a *prescribed* effective mapping $\varphi: Z \longrightarrow d(f)$.
6. There exists a similar mapping $\psi: r(f) \longrightarrow Z$.

These data are now necessary and sufficient to allow us to construct a diagram of mappings analogous to that characterizing the behavior of the Turing machine:

$$\begin{array}{ccc} d(f) & \xrightarrow{f} & r(f) \\ \varphi \uparrow & & \downarrow \psi \\ Z & \xrightarrow[\bar{f}]{} & Z \end{array} \qquad (2)$$

Here, as usual, $\bar{f}$ is the numerical function induced by the above data; $\bar{f} = \psi f \varphi$.

It is readily seen that any numerical function induced by an agent f satisfying the data 4 to 6 above is in any intuitive sense effectively calculable. Conversely if $\bar{f}$ is any numerical function whose values can be calculated by any sort of physical process whatsoever, then this process must at the very least involve the operation of an agent f satisfying 4 to 6, such that a diagram of the above type can be constructed. That is, whatever else may be involved, the bare minimum requirements of an effective incoding of integers, the operation of the agent, and the effective decoding of the outputs of the agent must be satisfied. This argument leads us to the following rigorous definition: a numerical function $\bar{f}: Z \longrightarrow Z$ will be called *physically effectively calculable* if and only if there exists an agent f and effective mappings φ, ψ satisfying 4 to 6 above, such that the function $\bar{f}$ is induced via diagram (2) by the operation of f.

It should be noted that we have deliberately restricted our attention to agents which receive *real* physical inputs. There exist other types of devices, which intuitively may be considered as capable of calculating numerical functions, but which do not require the integers to be coded into any real input objects. As an illuminating example of such a device, we can consider the following construction, which was suggested to the author by Professor G. Y. Rainich: Let us suppose that a fragment of radioactive material is placed within an insulated box. The box is provided with an aperture which determines a certain solid angle θ with the radioactive material as vertex. Let us suppose the material in question to be chosen so that, on the average, one particle per minute (say) is emitted within the solid angle θ. We further suppose that there is a scintillation screen set at a fixed distance from the aperture, which is divided into halves, with some convention established in advance to insure that no ambiguity results from a particle striking the screen on the line of division. Finally, we suppose that the aperture is provided with a door which is opened for an instant once each minute by a clockwork mechanism. We define a numerical function f as follows: $f(n) = 0$ if no scintillation appears on the screen when the aperture is opened at the nth minute; $f(n) = 1$ if a scintillation appears in the left half of the observing screen; $f(n) = 2$ if the scintillation appears in the right half of the screen. To avoid the trivial objection that this function ultimately becomes identically zero, we suppose that the radioactive material is replaced at intervals by a fresh sample; for instance, whenever its half life is exceeded. Devices of this type are not calculating agents, in the sense of our present discussion, and hence are excluded from our subsequent considerations.

IV. *The Description of Physical Systems*. We must now discuss briefly the manner in which the physicist mirrors the activity of real physical systems or processes by purely mathematical means. In physics, a system or process may be said to be *described* if there exists an algorithm which enables us to compute, from a suitable set of data characterizing the state of the system at some initial time of reference, a (maximal) amount of information about the state of the system at any later time. Algorithms of this type are referred to in physics as *laws*. We need not dwell here on how such algorithms are constructed, except to remark that they are

obtained by generalizing, in some sense, the results of suitable empirical observations. We should further remark that not all the statements which are presently called "physical laws" are of this *algorithmic* type (e.g., the laws of thermodynamics). In what follows, we restrict our attention entirely to those physical laws which enable us to determine future states of a system from some initial state; clearly the laws of thermodynamics and similar statements do not fall into this category.

There exist, even at the present time, certain classes of physical processes for which the physicist feels that he has already succeeded in formulating a set of algorithms which are optimal, in the sense that *any* system belonging to the class in question can be described in terms of these algorithms. Thus, for example, von Neumann (1950) says of the quantum-mechanical formalism: "...the theory, so far as it deals with individual electrons or with electron shells of atoms or molecules, is entirely successful, as it is also whenever it deals with electrostatic forces and with electromagnetic processes connected with the production, transmission, and transformation of light". A similar assertion may be made with regard to the efficacy of the Hamiltonian formulation of Newtonian mechanics, with regard to the description of macroscopic mechanical systems moving slowly compared with the velocity of light.

On the other hand, there are also many classes of physical processes for which suitable algorithms have not as yet been forthcoming. This of course does not imply that such algorithms *cannot* be constructed. In fact, the structure of theoretical physics is often asserted to contain the following two hypotheses, which imply that just the contrary is true:

> *Hypothesis I.* Every possible physical process admits of a description; i.e., a definite mathematical procedure whereby all future states of the process may be calculated from suitable initial data.
>
> *Hypothesis II.* Only a *finite* set of algorithms (i.e., laws) is required in order to provide a description of every possible physical phenomenon.

The first of these hypotheses may be regarded as asserting that physical science is in principle completely effective with respect to its ability to describe the external world. With regard to the

second hypothesis, we may point out that many physicists (including Einstein) have asserted a much stronger hypothesis; namely, that a *single* principle would be found from which a description of every possible physical system could be derived. One of the consequences of these hypotheses, which will be used below, is that they allow us to meaningfully use the phrase, *the laws of physics,* even though physics is at present far from a closed subject.

It must of course be remarked that there are many physicists who will deny either *Hypothesis I* or *Hypothesis II*. For example, Bohm (1957) has explicitly put forward the hypothesis that a countable infinity of different algorithms is required for the description of all physical processes, thereby denying *Hypothesis II*. However, Bohm accepts *Hypothesis I*. Bohm points out, however (*ibid.*, p. 84 *et seq.*) that many quantum physicists will deny *Hypothesis I*, as well, citing such processes as the decay of a single radioactive nucleus as an example of a physical process for which it is claimed that no description can ever be found. We shall analyze these views somewhat further in Section VI below, and retain *Hypotheses I* and *II* throughout the ensuing argument.

It may not be out of place to remark at this point that *Hypotheses I* and *II* above bear a certain resemblance to Hilbert's program for the formalization of mathematics. This may be seen by replacing the term *law of physics* by the term *postulate,* and replacing *description of a physical system* by *theorem*. This similarity between the assumptions implicit in the structure of theoretical physics and the Hilbert program is perhaps to be expected in view of Hilbert's deep familiarity with the physical sciences and his views on the origin of the basic concepts of mathematics. At any rate, in view of the subsequent fate of the Hilbert program, it may be well to bear this analogy in mind as we proceed with our discussion.

V. *The Turing-Computability of the Numerical Function Induced by an Arbitrary Calculating Agent.* We now proceed to the proof of the main result of the paper; namely, given *Hypotheses I* and *II*, *every physically effectively calculable function is Turing-computable*. We shall show that, given an arbitrary agent, a procedure whereby its associated numerical function can be Turing-computed may be derived from the physical description of the agent in question.

Let us suppose that a physical entity f is capable of acting as a calculating agent. According to our discussion in Section II above, this means that we must be able to construct the diagram of mappings

$$\begin{tikzcd} d(f) \arrow[r, "f"] & r(f) \arrow[d] \\ Z \arrow[u] \arrow[r, "\bar{f}"'] & Z \end{tikzcd}$$

where the properties of the various symbols are as described in Section II. Since φ is effective, we can by definition identify for each integer $n \varepsilon Z$ a definite and unique object $\varphi(n) \varepsilon d(f)$. Moreover, both f and $\varphi(n)$ are definite physical entities, and hence, by *Hypothesis I* of Section III, they admit of a physical description. In fact, we can see intuitively that the states of both f and $\varphi(n)$ will remain constant in time when these objects are considered in isolation.

Next, let us suppose that at some initial time t_0 the object $\varphi(n)$ is supplied to the agent f as an input. We thus arrive at what we have agreed earlier to call a calculating system. Intuitively, we see that the effect of the calculating system is to transform in some manner the original input object $\varphi(n)$ into a definite output object in $r(f)$. That is, the state of the calculating system, which we may denote by $f \oplus \varphi(n)$, is not constant in time.

Invoking *Hypothesis I* for the system $f \oplus \varphi(n)$, we see that we can effectively describe this system; i.e., there exists a definite (mathematical) procedure whereby we may determine the state of the system at any future time when we are given a suitable set of data concerning the state of the system at time t_0. In particular, we can determine whether or not, at some time subsequent to t_0, the process ceases (i.e., its state becomes constant). The cessation of the calculating process intuitively corresponds to the termination of the calculation; physically this means that the system now consists of two independent stable parts. One of these parts is by definition an object in $r(f)$, which we may denote by $f[\varphi(n)]$; the other part is the agent f with which we started (perhaps in a different state, corresponding to what in the Turing machine is a change in internal configuration). Moreover, these parts are identifiable from the description of the final stable state of

the calculating system $f \oplus \varphi(n)$; in particular, we can identify the object $f[\varphi(n)]$.

It might be objected at this point that the determination of whether the state of the system $f \oplus \varphi(n)$ ultimately becomes constant or not is not necessarily an effective process. However, we are at present dealing by hypothesis with the equations of motion of a physical system, which are quite highly determined, so that in practice it is quite possible to tell whether the cessation of interaction is permanent or not. One has an analog of this in the quite trivial problem of asking how one can be sure that a 4 will never appear in the decimal expansion of 1/3.

Once we have arrived at the object $f[\varphi(n)] \varepsilon r(f)$, we can by the effectiveness of the mapping ψ associate this object with a unique integer. This integer, by definition, is $\bar{f}(n)$. Of course, if the calculating system $f \oplus \varphi(n)$ never attains a constant state, then $\bar{f}(n)$ is left undefined.

We must now show how the various physical descriptions we have invoked above may be used to construct a Turing machine which will compute $\bar{f}$. It will be helpful to consider first a particular example, which will clearly illustrate the procedure to be followed in the most general situation. Accordingly, let us for the time being suppose that the agent f under consideration is completely describable by the laws of classical mechanics, which as we remarked above are regarded by physicists as being, in a sense, optimal. We suppose once more that at time $t = t_0$, a particular object $\varphi(n) \varepsilon d(f)$ is supplied to f as input. According to the formalism of classical mechanics (see Joos, 1950, for a concise exposition), the system $f \oplus \varphi(n)$ is completely described by its Hamiltonian function $H(p_1, \ldots, p_k, q_1, \ldots, q_k)$, where the q_i are appropriate generalized co-ordinates and the p_i are the associated momenta. The change of state of this system with time is given by the set of $2k$ simultaneous differential equations

$$\begin{cases} \dfrac{dp_i}{dt} = -\dfrac{\partial H}{\partial q_i} \\ \dfrac{dq_i}{dt} = \dfrac{\partial H}{\partial p_i} \end{cases} \quad i = 1, \ldots, k$$

subject to the initial conditions obtained by specifying the $2k$ numbers $\{p_i(t_0), q_i(t_0)\}$. These equations are called the ***equations of motion*** of the system.

The equations of motion, whether or not they are directly solvable in closed form, can in principle always be solved with the assistance of the ordinary techniques of numerical computation. For example, the $2k$ differential equations may, by the use of standard relaxation methods, be reduced to difference equations, and the solutions of these difference equations may be made to approximate to the solutions of the actual equations of motion to any accuracy desired by choosing a suitably fine mesh. If we approximate to the true solutions of the equations of motion sufficiently closely, there will be no physically detectable difference between the behavior of the system as predicted by the approximate solution and the actually observed behavior of the system.

We now observe that these numerical techniques may be performed on a properly programmed digital computer. Since it is known that every such computer can be imitated by an appropriate Turing machine, it is already possible for us to conclude that the numerical function $\bar{f}$ induced by the agent f under consideration must be Turing-computable.

More specifically, we see that the power of the equations of motion is to enable us to mirror the physical activity of the calculating system $f \oplus \varphi(n)$ by a series of strictly mathematical, and, in fact, strictly numerical procedures. Since in the last analysis it is this activity which defines the numerical function $\bar{f}$, we have thereby succeeded in associating with $\bar{f}$ a series of processes which can be carried out on a suitable Turing machine, and which will actually compute $\bar{f}$. We can now write down, in a completely explicit fashion, the various steps involved in the transition from the agent f to an appropriate Turing machine, as follows:

1. Given an integer n, we determine the object $\varphi(n)$.
2. To each object $\varphi(n)$, we associate the equations of motion of the system $f \oplus \varphi(n)$. This enables us to compute the state of this system at any time $t > t_0$, given appropriate information about f and $\varphi(n)$ at time t_0.
3. The equations of motion are next transformed into a suitable numerical form, so that the transformed equations are (a) solvable on a suitable digital computer, and (b) sufficiently close to the actual solution so that no detectable physical consequence results from using the approximation.
4. The state of the system $f \oplus \varphi(n)$ at a convenient sequence of times (say $t_0 + 1$, $t_0 + 2, \ldots$) is determined by means of

the approximation selected in (3) above, in order to ascertain whether or not the state of $f \oplus \varphi(n)$ ultimately becomes constant.

5. If we find that, subsequent to a certain time $t_0 + N$, the system $f \oplus \varphi(n)$ maintains a constant state, then we can obtain a description of that state by substituting $t_0 + N$ into the Hamiltonian of the system.
6. The description of the constant state obtained in the preceding step is by hypothesis directly decomposable into a description of f and a description of the appropriate object $f[\varphi(n)] \in r(f)$. In other words, this constant state uniquely and effectively determines the object $f[\varphi(n)]$.
7. By means of the mapping ψ, we can associate a unique integer with the object $f[\varphi(n)]$. This integer is by definition $\bar{f}(n)$.

We can represent this procedure schematically, in the following manner: suppose that T is a Turing machine which can carry out the computations specified by steps 3 and 4 above. As we saw in Section II, T induces a mapping Φ from the free semigroup $G(A)$ generated by the alphabet of the machine into itself. Since T is already a Turing machine, we are by hypothesis given appropriate mappings $\varphi' : Z \longrightarrow G(A)$, $\psi' : G(A) \longrightarrow Z$ (in fact, in this case we have $\psi' = (\varphi')^{-1}$). Then we can construct the following diagram of mappings:

$$\begin{tikzcd}
G(A) \arrow[rrr, "\Phi"] \arrow[dr, "\rho"'] & & & G(A) \arrow[ddl, "\psi'"] \\
& d(f) \arrow[r, "f"] & r(f) \arrow[ur, "\theta"'] \arrow[d, "\psi"] & \\
& Z \arrow[u, "\varphi"] \arrow[uul, "\varphi'"] \arrow[r, "\bar{f}"] & Z &
\end{tikzcd}$$

where the mappings ρ and θ can obviously be defined effectively in such a manner as to render the entire diagram commutative. The computability of $\bar{f}$ is now obvious, using the definition $\bar{f} = \psi' \Phi \varphi'$.

We now return to a consideration of the general calculating agent f. We first remark that the special example which we have just considered in detail is in fact perfectly typical of how the laws of physics, and the description of particular physical systems

which can be derived from them, can be used to construct specific algorithms whereby associated numerical functions can be computed in the sense of Turing. It may be pointed out that all the laws of physics thus far discovered have taken the form of differential equations, and for these the discussion supplied above may be carried over almost verbatim. But even if it should be the case that sometime in the future a physical law should be discovered which is not of this form, it is nevertheless clear that only minor details of the discussion will be altered.

We may thus sum of the results of this section in the following form: If *Hypotheses I* and *II* are correct, then it follows (at least informally) that Church's Thesis, considered as a physical proposition, is a true statement. (In fact we have thus far only utilized *Hypothesis I*, but since *Hypothesis II* implies *Hypothesis I*, we can repeat the above arguments on the basis of *Hypothesis II* as well). Contrapositively, we have shown that if the physical form of Church's Thesis is false, then *Hypothesis I* and *II* cannot both be true, and may very well both be false.

VI. *Discussion.* Before proceeding to investigate the consequences of the falsity of Church's Thesis, let us consider briefly the various alternatives to *Hypotheses I* and *II*. We have considered these hypotheses separately because, as we have seen in Section IV, it is possible to deny *Hypothesis II* alone, or to deny both hypotheses.

The denial of *Hypothesis II* is equivalent to asserting that physics is essentially *incomplete*, in much the same sense that Gödel showed arithmetic to be incomplete. That is, given any finite set of physical laws, there exist systems which are not describable in terms of these laws. The denial of *Hypothesis I* asserts in effect that physics is *essentially incomplete*, in a sense far stronger than the Gödel-incompleteness of arithmetic. An arithmetic analog of this type of incompleteness would entail the existence of some arithmetic proposition which does not follow from *any other* proposition or set of propositions which are mutually consistent.

If it is assumed a priori that *Hypothesis I* is false, we see that the physical form of Church's Thesis becomes *undecidable*. If we are given a calculating agent f for which no description can be found, there is no means of determining whether the associated numerical function $\bar{f}$ is Turing-computable or not. For example,

under the hypothesis that the radioactive decay of an atomic nucleus is essentially undescribable, we might be able to construct a true computing agent for which it is impossible to decide whether or not its associated numerical function is Turing-computable or not.

If *Hypothesis II* alone is denied a priori, then we can still formally reach the conclusion that the physical form of Church's Thesis is a true statement by following the argument of the preceding section. However, in these circumstances such an argument is no longer *effective*, in a logical sense.

It is at any rate clear that, if it should be the case that Church's Thesis is false as a physical assertion, then *Hypotheses I* and *II* cannot both be correct. While our purpose has been primarily to show how the Thesis is related to the laws of physics, with the goal of ultimately obtaining a possible characterization of realizability, and not to investigate the validity of *Hypotheses I* and *II* as foundations for theoretical physics, we might point out that there is no real analog in physics of the long list of unproved propositions in elementary number theory, which resisted all attempts at proof for so long that it at last became plausible to investigate the possibility that some of them were actually unprovable.

One final possible biophysical implication of our discussion may be mentioned. If the rational properties of the mind are regarded as a manifestation of a complicated physical system which is capable of functioning as a calculating agent, then it follows that every calculating scheme which the mind can devise can, if Church's Thesis in its physical form is correct, be carried out by a Turing machine. Furthermore, since every scheme for effectively calculating the values of a numerical function is an outcome of a mental effort, we may very well come close to providing a proof of the Church Thesis in its original form.

The author is indebted to Professor N. Rashevsky, Professor G. Y. Rainich, and Professor Patrick Suppes for valuable discussions concerning the subject matter of the above research.

LITERATURE

Bohm, D. 1957. *Causality and Chance in Modern Physics.* London: Routledge & Kegan Paul Ltd.

Bremermann, H. J. 1962. "Optimization Through Evolution and Recombination." In *Self-Organizing Systems*, M. Yovits, G. T. Jacobi, and G. D. Goldstein, eds. Washington: Spartan Books.

Church, A. 1936. "An Unsolvable Problem of Elementary Number Theory." *Am. J. Math.*, 21, 275–281.

Davis, M. 1958. *Computability and Unsolvability.* New York: McGraw-Hill Book Co.

Joos, G. 1950. *Theoretical Physics.* New York: Hafner Publishing Co.

Kleene, S. C. 1952. *Introduction to Metamathematics.* New York: D. Van Nostrand Co., Inc.

von Neumann, J. 1950. *Mathematical Foundations of Quantum Mechanics.* Princeton: Princeton University Press.

Rashevsky, N. 1954. "Topology and Life." *Bull. Math. Biophysics*, **16**, 317–348.

Riemann, B. 1892. "Uber die Hypothesen, Welche der Geometrie zu Grunde Liegen." Mathematische Werke (2 Aufl., Nr. XIII, p. 272). Leipzig.

Rosen, R. 1958a. "A Relational Theory of Biological Systems." *Bull. Math. Biophysics*, **20**, 245–260.

———. 1958b. "The Representation of Biological Systems from the Standpoint of the Theory of Categories." *Ibid.*, **20**, 317–341.

———. 1959. "A Relational Theory of Biological Systems II." *Ibid.*, **21**, 109–128.

———. 1961. "Recent Progress in Relational Biology." In *Proceedings of the Cullowhee Conference on Training in Biomathematics.* Raleigh: North Carolina State College (to appear).

———. 1962. "A Note on Abstract Relational Biologies." *Bull. Math. Biophysics*, **24**, 31–38.

Turing, A. M. 1937. "On Computable Numbers." *Proc. London Math. Soc.*, **42**, 230–265.

RECEIVED 5-28-62

UNCLASSIFIED

UNCLASSIFIED